VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES

Jinrong hu

Badminton Hall Reservation System

Unit: Technology
2021

Acknowledgments

I would like to thank the many people who have helped me through the process of completing my thesis.

Firstly a great thanks to my supervisor Ghodrat Moghadampour, without his patient teaching and encouragement when I was at a low point, I may have found it difficult to keep going through.

Secondly thanks to my friends Haoyuan Sun and Jingyu Liu who told me some experiences when I was writing my report.

Lastly, I would like to thank my family for their support, without them I would never have had the chance to complete my studies.

Jinrong Hu

VAASAN AMMATTIKORKEAKOULU
UNIVERSITY OF APPLIED SCIENCES
Information Technology

ABSTRACT

Author	Jinrong Hu
Title	Badminton Hall Reservation System
Year	2022
Language	English
Pages	45+10 appendices
Name of Supervisor	Ghodrat Moghadampour

This project is an online badminton booking system where users become members by registering on the website and members can make badminton court bookings online. This is a good way to avoid the complicated operation of manual telephone booking. And the news section on the website also allows customers to better understand badminton court information and badminton related news.

The design and production of the badminton court booking system uses an integrated framework of Spring,Struts and Hibernate, a web app developed using the Java programming language. Struts is used as the base framework for the entire system, managing the separation of the MVC model and the jumping of JSP pages. Hibernate is the persistence layer framework for this system, managing the mechanism for processing data. Spring acts as the manager of the whole system, managing Struts and Hibernate.

The system has three user-facing roles, each with a different function. The first role is User, which is the basic user facing role. This role can become a member of the system through the registration function. After becoming a member, the user can use the booking function to book a badminton court online. The user can also browse the pages of the website and view the news section for enquire, but cannot add or modify them. The second role is operator, which can also be called employee. Accounts for this role cannot be registered on their own, but need to be assigned by the administrator. After logging into their account, employees can edit the information on the badminton court and make changes to the information in the news section. The employee can also make changes to the balance of the member's account after receiving payments. The final role is that of administrator. This role has the highest authority in the system. The administrator account has all the functions of the employee account and can make changes to the equipment function, the coach function and the rotating image function of the web page.

Keywords Badminton court reservation system, Java,Spring,Struts,Hibernate.

CONTENTS

LIST OF FIGURES AND TABLES

LIST OF APPENDICES

1 INSTRUCTION

The web application has a number of different advantages. The first advantage is that it does not require any installation and can be accessed via a browser. The second advantage is that web applications are cross-platform and multi-device, users can access whatever software they want from any device, a computer, a tablet, a smartphone too. Finally, it is very adaptable, visually intuitive and very easy to update when necessary. (computertechreviews.com,online)

1.1 Motivation

Traditional badminton hall usually relies on manual work to collect, collate, modify and store the information, which consumes a huge amount of human, material, and financial resources, and cannot afford the ever-expanding booking information. Therefore, with the rapid development and popularization of computers and the Internet, many companies and users have resorted to network management systems.

1.2 Objective

Considering the large quantity and variety of the booking options, it is essential to create a management system with larger storage. Developing a suitable badminton hall booking system will make it easier for users to view information about badminton hall bookings and thus improve the efficiency of managing badminton hall booking information.

Users can register as a member using the website. Members can see the badminton court information on the website. Members can book a court online, view news etc. Administrators can manage membership information and venue information, etc. Cashiers can recharge customers and manage orders etc.

This design provides the system analysis, requirement analysis, design analysis and functional analysis of the badminton booking system. The overall planning and design have been carried out in terms of development background, development environment, objectives, processes, database and system maintenance. Java technology and MySQL database are used to ensure the stability and development of the system. The badminton court reservation system makes the information management of badminton courts more systematic, standardized and efficient. (Lie, Chen 2018)

In this system, the functions that the system must have are online registration function and online booking function. The system should also have a news function, and a preview function for coaches and equipment. The best features to have in this system are a rotating image function on the homepage and an online payment function.

2 RELEVANT TECHNOLOGIES

The badminton court booking system adopted a MySQL database and Java technology, with a focus on the design and operation of the database. The combination of Java and MYSQL was to ensure the feasibility and effectiveness of the development of the booking system.The MySQL database has been chosen to be used in this system because firstly it is very simple to operate and the interface looks very clean. The second is that for this small amount of data using msysql would be more convenient than using an oracle database. The third point is that it is free of charge.

Java websites as well as the Tomcat web server are fully supported by most programs in the Windows operating system. Websites for small to medium sized platforms match well with Windows and Tomcat's powerful free software or hardware requirements. (Tomorrow's Technology 2018)

2.1 Tomcat

Tomcat Server is a free and open source web application server. It is a lightweight application server that is commonly used in small to medium sized systems and where there are not many concurrent users, and is the first choice for developing and debugging JSP programs. (Baidu.com online)

2.2 Dao

DAOs (Data Access objects) are objects that sit between business logic and persistent data to enable access to persistent data, which makes it the encapsulation of all database operations. (runoob.com online)

2.3 LayUI

LayUI is a front-end UI framework written in its own modular specification, following the native HTML/CSS/JS writing and organization format, with very low barriers to its entry. It mainly provides many well-designed styles and is easy to use. Bootstrap provides similar functions, but its framework is more advantageous in defining various front-end interaction style interfaces, such as the paging form which only configures the interface in the front-end with the back-end in accordance with the defined interface rules to return data, thus greatly reducing the development costs of back-end personnel when completing the page display. (segmentfault.com online)

2.4 JDBC

JDBC (Java Database Connectivity) is an application programming interface in the Java language used to standardize client programs' access to databases and provide methods such as querying and updating data in the database. JDBC is also a trademark of Sun Micro-systems. (Baidu.com online)

2.5 MySQL

MySQL is a powerful and efficient database server that can be accessed by multiple users at the same time, possesses multiple threads to process transactions, and secures a relational database system. A key feature of the MySQL database is that it is free and open source. Due to its open-source nature, it can be redeveloped by all programmers across the world, making MySQL databases more diversified, versatile, simpler and easier to use. Each database has its own interface and MySQL's data interface can match many of the popular languages on the markets, enabling the true multi-user and multi-threading experience. In addition, according to a survey by the world software organization IIEDs, the MySQL database is currently the most used database for small to medium sized program development and is one of the most preferred by programmers because of its open source and ease of use. (He 2019)

3 APPLICATION DESCRIPTION

The Badminton Court Booking System is a web application that allows users to register online for a membership account and book a court. The operator can manage all members' accounts and make changes to the balance and personal information in the account. The operator can also change the venue information, equipment and instructor status. The administrator will add three new functions to the operator's rights, the first function is the distribution of the operator's account. The second function is the modification of rotating images and the third function is the addition or modification of the news section.

3.1 Requirement's Analysis

The overall design of the system architecture is a process of subdividing a large task into several smaller tasks which, when completed in sections, are combined together to form a complete task. It works in the following specific steps.

1> The system is broken down into multiple sub-modules

2> The functions of each sub-module are pre-designed

3> the logical relationships between the sub-modules are designed

4> the design of the interface of each module and the transfer of information between modules

The overall structure of the badminton court booking system is designed as shown in the figure 1.

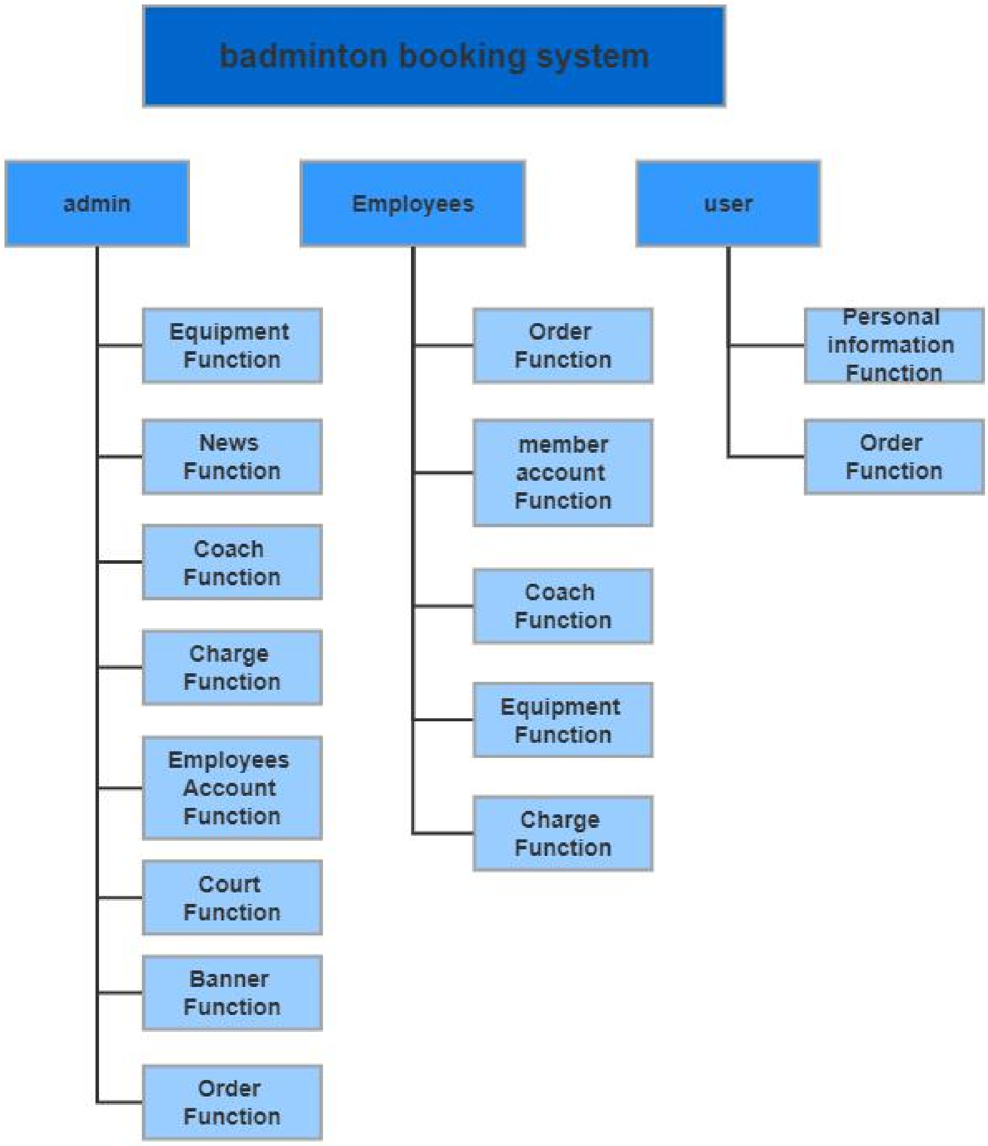


Figure 1. Overall structure.

3.1.1 Administrators' Function Requirements

Administrators enter the system through the administrator login screen and jump directly to the back office administration screen. The administrator can add, delete, modify and query the content of the News section, Coaching section, Rotating images section, Badminton court section, Equipment section and Orders section. The administrator can check, modify, add and delete information on employee accounts. The administrator can make changes to members' account information and account balances. The administrator use case diagram is shown in Figure 2.

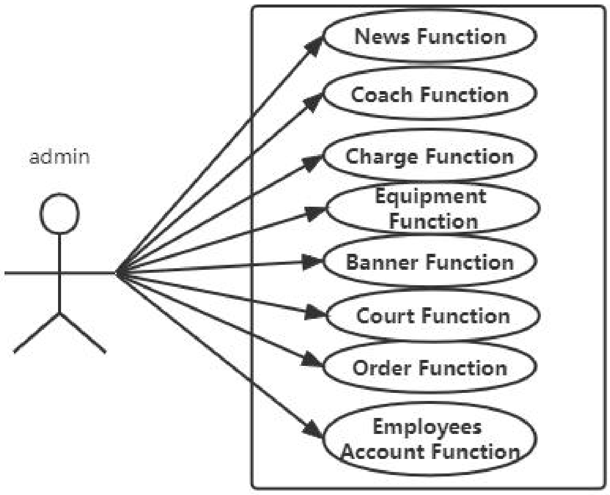


Figure 2. Administrator's functions.

3.1.2 User's Function Requirements

The user logs into the system via the user login screen and is stuck in the main page where they can view badminton court information, book venues, and news section content, but cannot edit or delete it. Once you have accessed your back office, you can view, delete and amend your personal information and order information as shown in Figure 3.

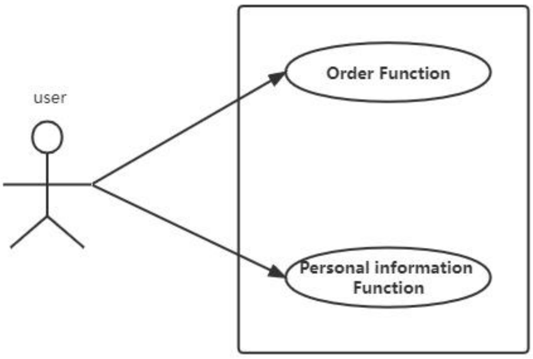


Figure 3. User's Function.

3.1.3 Employee's Function Requirements

The login screen for employees and administrators is the same, employees need to select their role as cashier and then enter their account information to login to the back office page. The employee can manage the order information, member's account information, equipment information and coach information in the back office. The employee can also add funds to the member's account. This means that the employee has the right to change the balance of the member's account as shown in Figure 4.

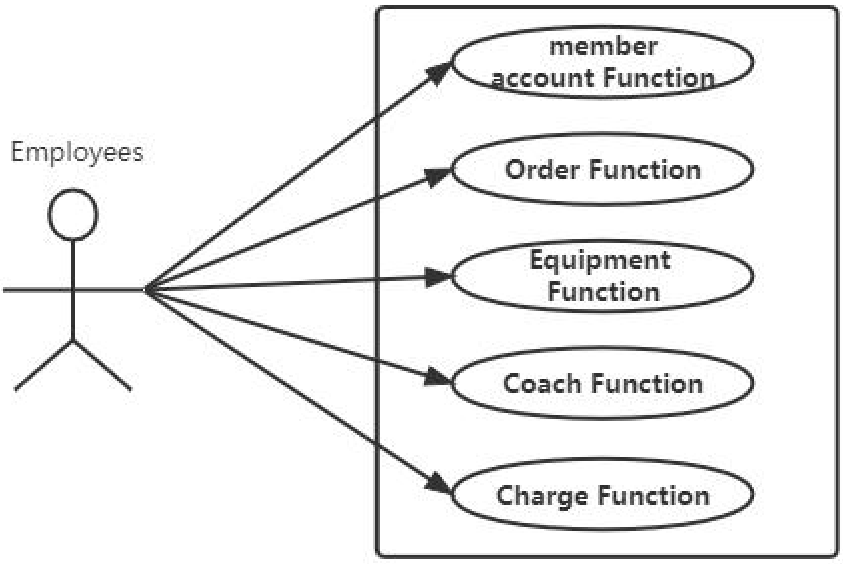


Figure 4. Cashier's Function.

3.2 Data Source Configuration Sequence Diagram

Sequence diagrams allow for better modularity of the system and clarify the functions of the individual modules. Below is a sequence diagram of all users performing login actions as shown in Figure 5.

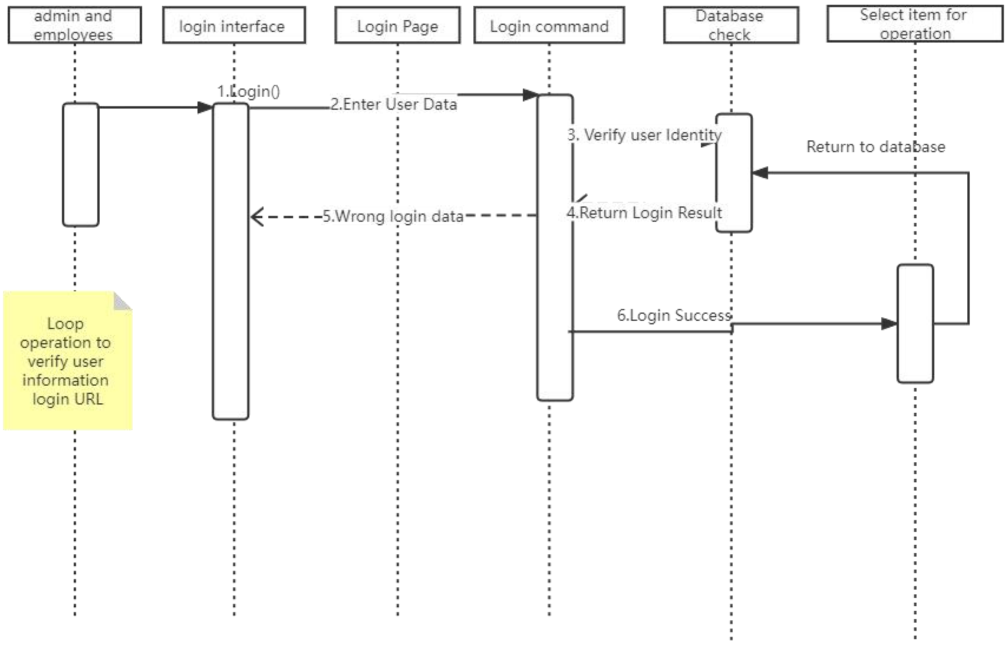


Figure 5. Login sequence diagram.

4 DATABASE

Databases are the basis of computer information systems. The collection, organization, storage, retrieval, updating, processing, statistics and dissemination of information must all be supported by a database management system. Currently, databases are the key and central part of computer systems. The quality of database development directly affects the quality and speed of the whole system. (Xiao 2017)

The design of a database can be divided into the following steps: requirements analysis, design concept, design logic and design physics. In conceptual design, there are usually four approaches. Top-down, bottom-up, incremental extension and hybrid strategies are used as skeletons to integrate the local conceptual structure removed in the bottom-up strategy. In the physical structure design phase, there are two steps. Determining the physical structure of the database, which in the case of relational databases is primarily the access method and storage structure. The physical structure is evaluated for time and space efficiency. Another step is selecting the appropriate access method for the relational schema. Common access methods include the access index method, the cluster access method and the hash access method. Generally, the design of the database is based on existing database management systems and MySQL, MySQL and Oracle. (Liu 2018)

The badminton court booking system uses the MySQL database management system and the details of the tables in the database are as follows.

Table 1. Admin administrator Information Table.

Field name	Data type	Default value	Nonempty allowed	autoincrement	Note
id	int(10)		NO	Yes	
username	varchar(50)		NO		username
pwd	varchar(50)		NO		pwd
addtime	timestamp		NO		addtime

Table 2. Equipment information table.

Field name	Data type	Default value	Nonempty allowed	autoincrement	Note
id	int(10)		NO	Yes	
Equipment name	varchar(50)		NO		Equipment name
Equipment picture	varchar(50)		NO		Equipment picture
Stute	int(10)		NO		status
Equipment introduction	decimal(18,2)		NO		Equipment introduction
add time	timestamp		NO		add time

5 IMPLEMENTATION

Based on the requirements of this project, the project architecture is designed in four layers: application layer, interface layer, service layer and database. In the app layer, HTML and CSS are employed to achieve static effects on the page and jQuery to achieve dynamic effects. Ajax is used to attain asynchronous refresh of the page. (Lin 2018)

In the interface layer, Com.action holds all the controllers in the back-end of the system, responsible for controlling the calls and page jumps of the Dao layer. Com.common holds all the public methods, including some operation classes on the database. Com.dao holds all the specific operations on the data in the system and the interfaces. Com.entity can be understood as the entity class, which is the definition of all the entities in the system. Com.entity defines the specific properties of the entities. The Dao layer of com.entity can be regarded as the entity class defining all the specific properties of the entity, the database fields, and primary keys for each property through annotations. Similar to the tool class, com.dao defines the methods for uploading files and threads in the system.

The service layer is equipped with three important functions and some technologies are used in the programming of the project. Finally, the processed data is stored in a database in the Storage Layer. The project's visual structure is shown in figure 6 below:

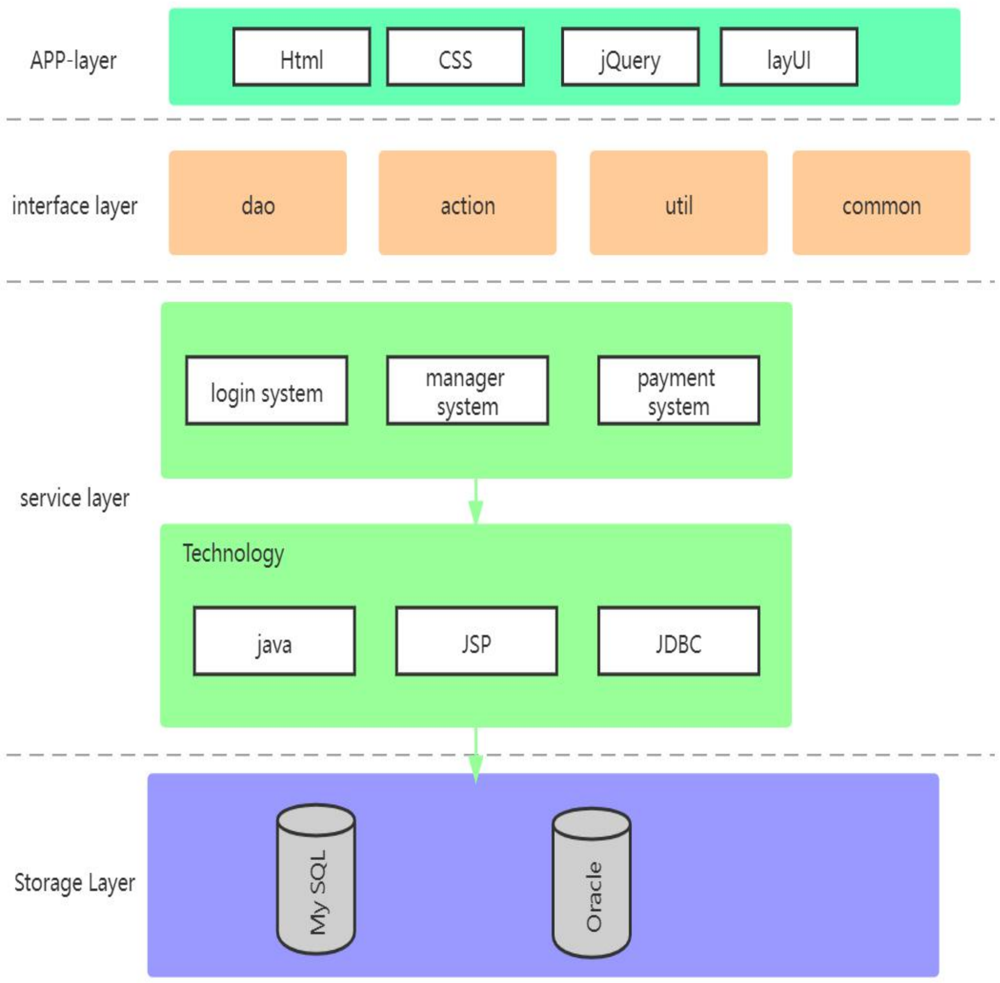
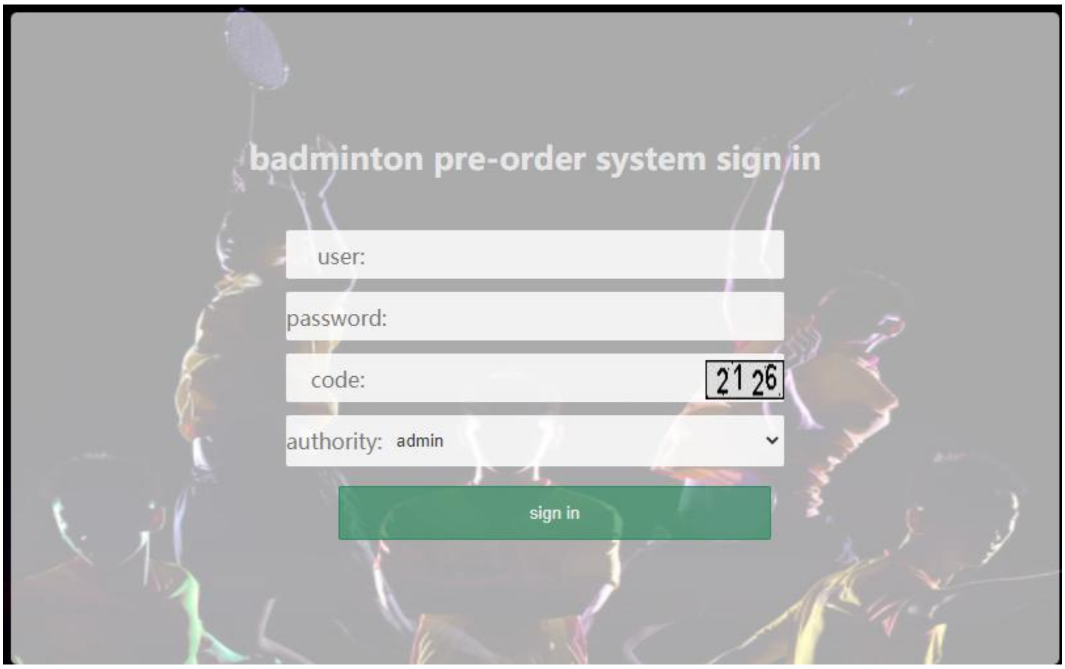


Figure 6. Project's vision structure.

5.1 System Login Interface

Upon entering the system, the first thing the user sees is the user login screen, then enter the account number and password and click Login to enter the corresponding main screen. The login screen is shown in Figure 7.



badminton pre-order system sign in

user:

password:

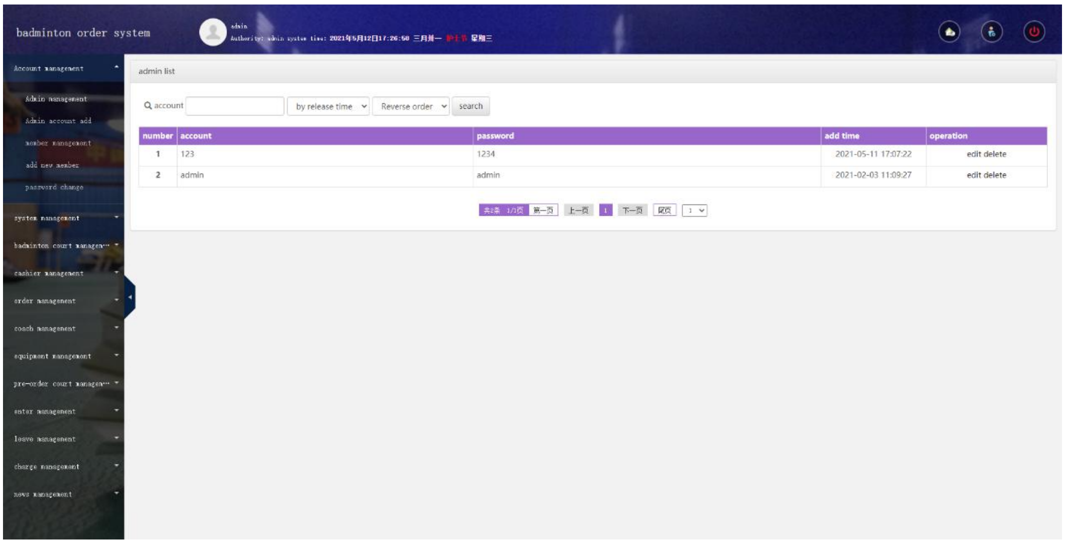
code: 2126

authority: admin

Figure 7. System login interface.

5.2 Account Management Interface

Once logged in, administrators can add, delete, modify and view administrator information, as shown in Figure 8.



badminton order system

admin
Authority: admin system time: 2021年5月12日17:26:50 三月卅一 护士节 星期三

admin list

account: by release time, Reverse order

number	account	password	add time	operation
1	123	1234	2021-05-11 17:07:22	edit delete
2	admin	admin	2021-02-03 11:09:27	edit delete

Figure 8. Account management interface.

5.3 System Management Interface

Administrators also have access to manage system information in the back office with the interface shown in Figure 9.

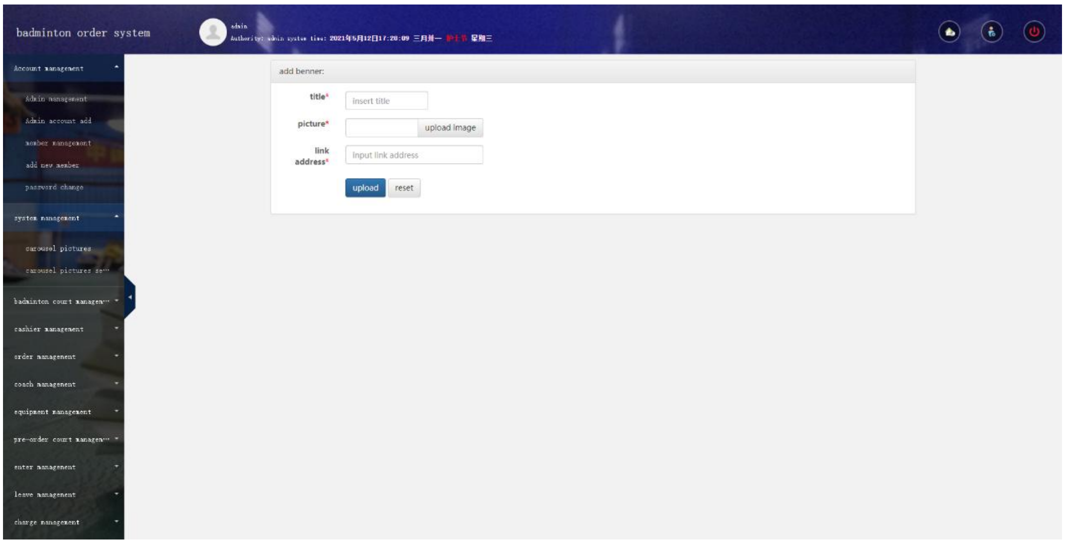
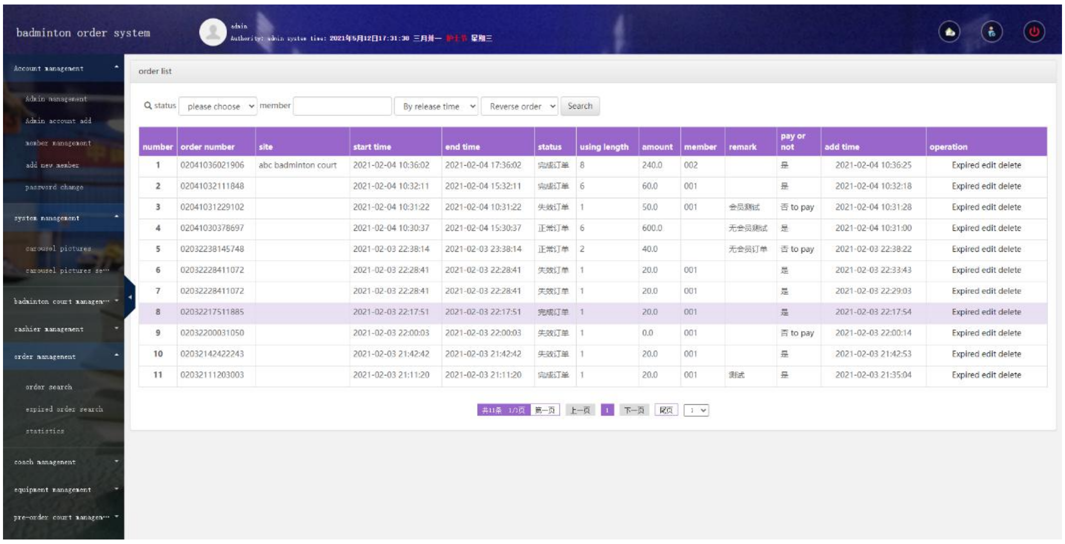


Figure 9. System management interface.

5.4 Order Management Interface

The administrator can view, add and manage the order information, and the interface is shown in Figure 10.



order list

status: please choose | member: | By release time | Reverse order | Search

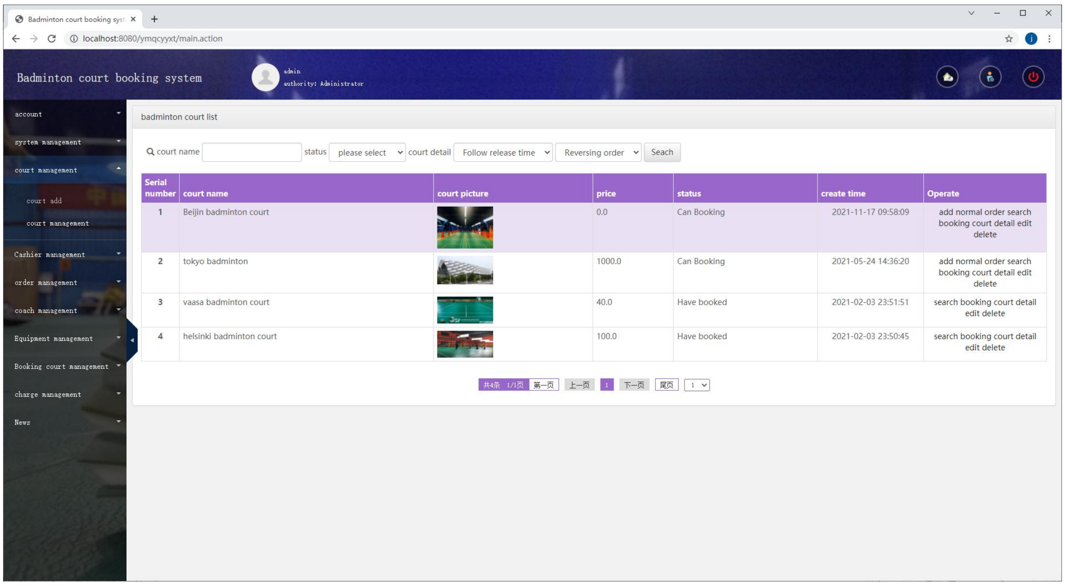
number	order number	site	start time	end time	status	using length	amount	member	remark	pay or not	add time	operation
1	02041036021906	abc badminton court	2021-02-04 10:36:02	2021-02-04 17:36:02	完成订单	8	240.0	002		是	2021-02-04 10:36:25	Expired edit delete
2	02041032111848		2021-02-04 10:32:11	2021-02-04 15:32:11	完成订单	6	60.0	001		是	2021-02-04 10:32:18	Expired edit delete
3	02041031229102		2021-02-04 10:31:22	2021-02-04 10:31:22	失效订单	1	50.0	001	会员测试	否 to pay	2021-02-04 10:31:28	Expired edit delete
4	02041030378697		2021-02-04 10:30:37	2021-02-04 15:30:37	正常订单	6	600.0		无会员测试	是	2021-02-04 10:31:00	Expired edit delete
5	02032238145748		2021-02-03 22:38:14	2021-02-03 23:38:14	正常订单	2	40.0		无会员订单	否 to pay	2021-02-03 22:38:22	Expired edit delete
6	02032228411072		2021-02-03 22:28:41	2021-02-03 22:28:41	失效订单	1	20.0	001		是	2021-02-03 22:33:43	Expired edit delete
7	02032228411072		2021-02-03 22:28:41	2021-02-03 22:28:41	失效订单	1	20.0	001		是	2021-02-03 22:29:03	Expired edit delete
8	02032217511885		2021-02-03 22:17:51	2021-02-03 22:17:51	完成订单	1	20.0	001		是	2021-02-03 22:17:54	Expired edit delete
9	02032200031050		2021-02-03 22:00:03	2021-02-03 22:00:03	失效订单	1	0.0	001		否 to pay	2021-02-03 22:00:14	Expired edit delete
10	02032142422243		2021-02-03 21:42:42	2021-02-03 21:42:42	失效订单	1	20.0	001		是	2021-02-03 21:42:53	Expired edit delete
11	02032111203003		2021-02-03 21:11:20	2021-02-03 21:11:20	完成订单	1	20.0	001	测试	是	2021-02-03 21:35:04	Expired edit delete

共11条 1/1页 第一页 上一页 1 下一页 尾页 1

Figure 10. Order management interface.

5.5 Badminton Field Management Interface

The administrator can view, add and manage badminton field information, and its interface is shown in Figure 11.



badminton court list

court name: | status: please select | court detail | Follow release time | Reversing order | Seach

Serial number	court name	court picture	price	status	create time	Operate
1	Beijin badminton court		0.0	Can Booking	2021-11-17 09:58:09	add normal order search booking court detail edit delete
2	tokyo badminton		1000.0	Can Booking	2021-05-24 14:36:20	add normal order search booking court detail edit delete
3	vaasa badminton court		40.0	Have booked	2021-02-03 23:51:51	search booking court detail edit delete
4	helsinki badminton court		100.0	Have booked	2021-02-03 23:50:45	search booking court detail edit delete

共4条 1/1页 第一页 上一页 1 下一页 尾页 1

Figure 11. Badminton field management interface.

5.5.1 Search code

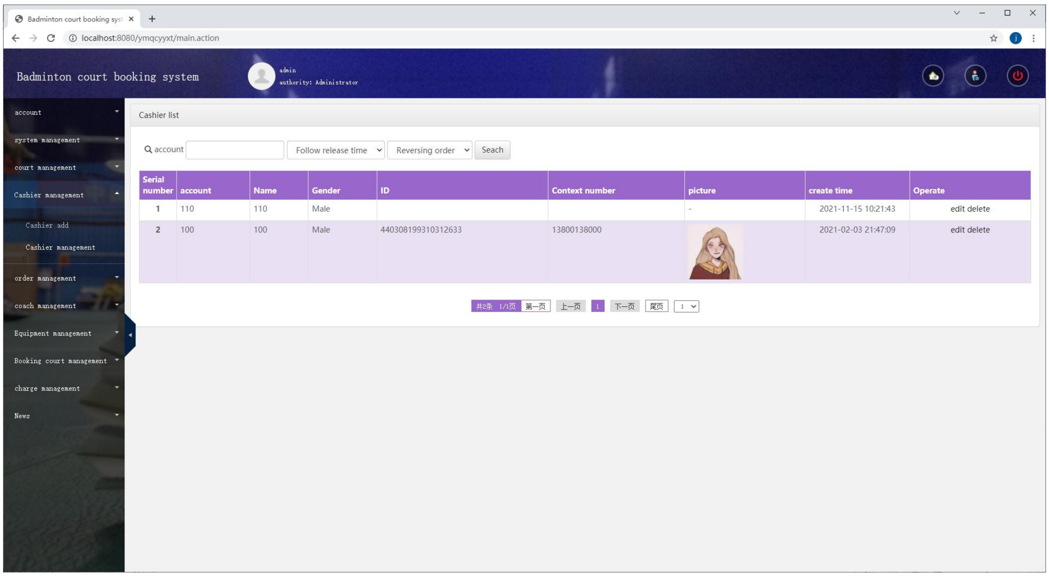
This code is applied to the search function in the program. The program sets the default id to be the inverted narrative and the default 15 data to be a page.

Code snippet 1. Index page date list.

```
public String index()

    {   String order = Request.get("order" , "id");

        String sort  = Request.get("sort" , "desc");

        int pagesize = 15;String where = " 1=1 ";where += 
getWhere();     long count = 
Long.valueOf(HbUtils.getCurrentSession().createQuery("SELECT 
count(*) FROM Yumaoqiuchangdi WHERE 
"+where).uniqueResult().toString()).longValue();

     SQLQuery 
query=HbUtils.getCurrentSession().createSQLQuery("SELECT * 
from Yumaoqiuchangdi WHERE "+where+" ORDER BY "+order+" 
"+sort).addEntity(Yumaoqiuchangdi.class);

        Collect collect = new Collect(count , pagesize);

            query.setFirstResult(collect.firstRow);

        query.setMaxResults(collect.listRows);

        List list = query.list();

        request.setAttribute("orderby" , order);

        request.setAttribute("sort" , sort);

        request.setAttribute("list" , list);

        return success;}
```

5.6 Cashier Management Interface

The administrator can view, add and manage the cashier information, and its interface is shown in Figure 12.

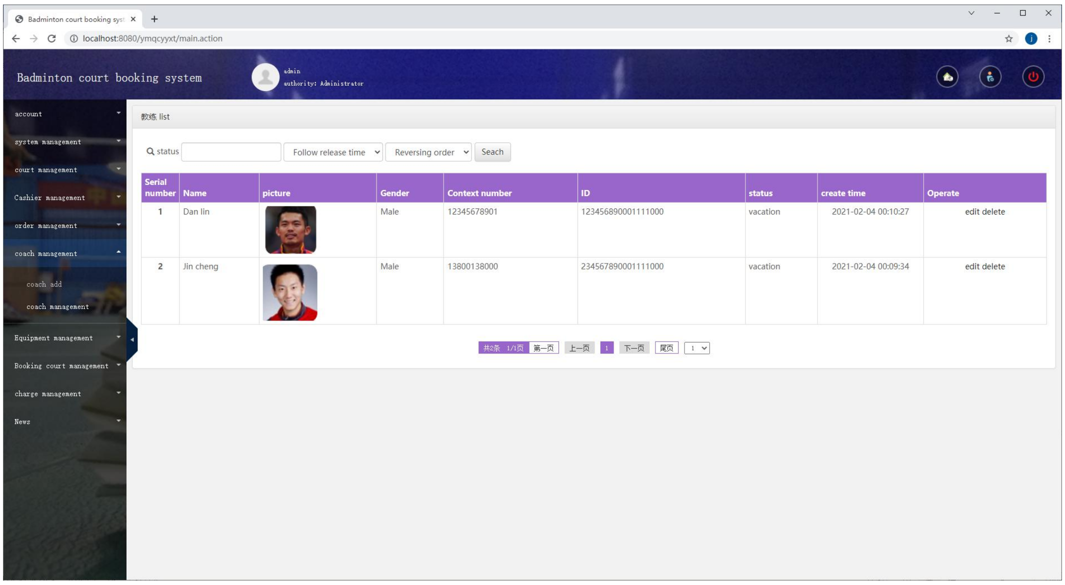


Serial number	account	Name	Gender	ID	Context number	picture	create time	Operate
1	110	110	Male			-	2021-11-15 10:21:43	edit delete
2	100	100	Male	440308199310312633	13800138000		2021-02-03 21:47:09	edit delete

Figure 12. Cashier management interface.

5.7 Coach Management Interface

The administrator can view, add and manage coach information, and its interface is shown in Figure 13.



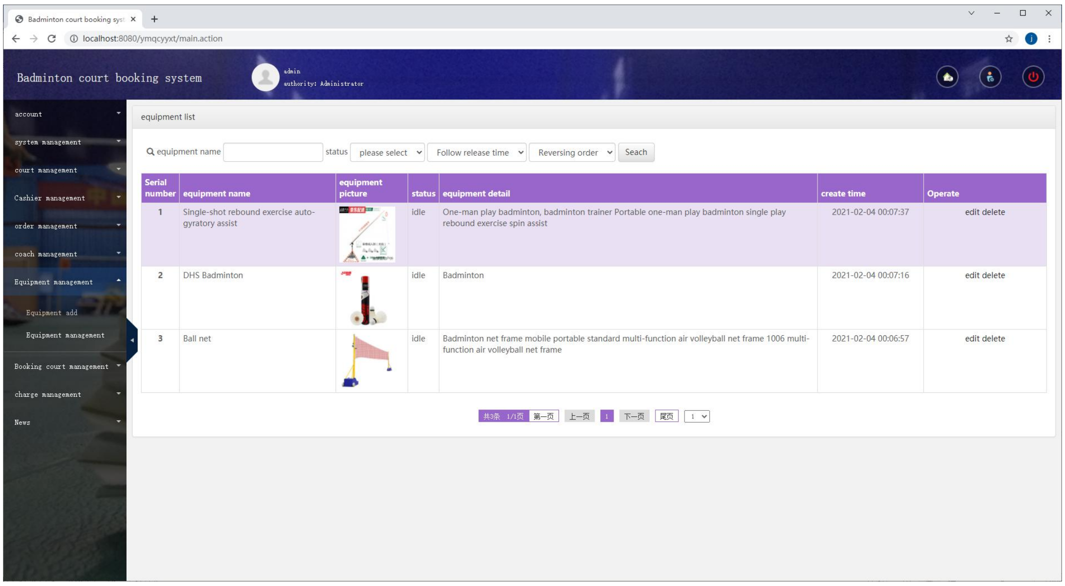
教练 list

Serial number	Name	picture	Gender	Context number	ID	status	create time	Operate
1	Dan lin		Male	12345678901	123456890001111000	vacation	2021-02-04 00:10:27	edit delete
2	Jin cheng		Male	13800138000	234567890001111000	vacation	2021-02-04 00:09:34	edit delete

Figure 13. Coach management interface.

5.8 Equipment Management Interface

The administrator can view, add and manage equipment information, and its interface is shown in Figure 14.



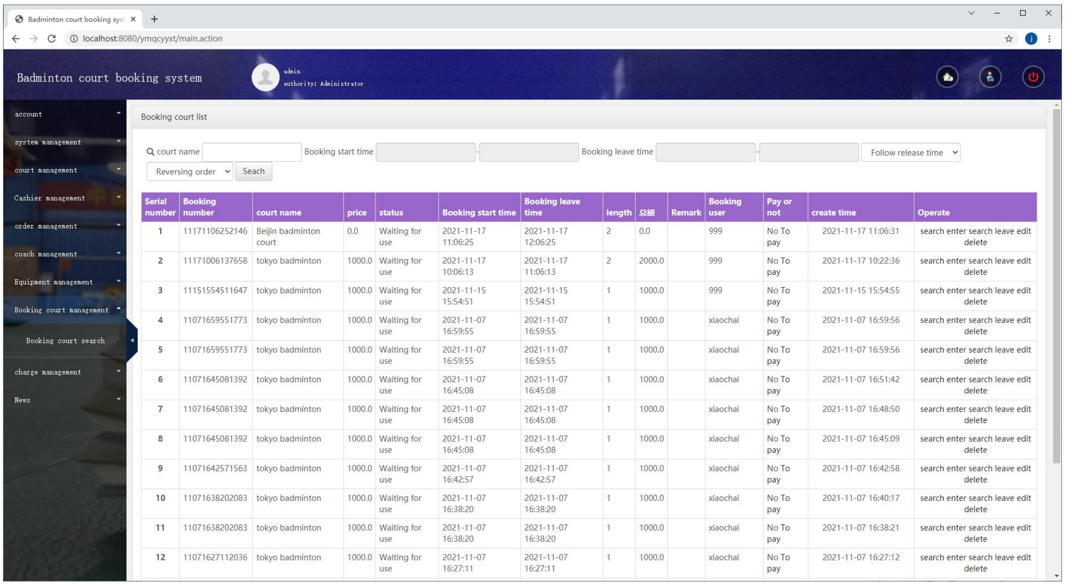
equipment list

Serial number	equipment name	equipment picture	status	equipment detail	create time	Operate
1	Single-shot rebound exercise auto-gyratory assist		idle	One-man play badminton, badminton trainer Portable one-man play badminton single play rebound exercise spin assist	2021-02-04 00:07:37	edit delete
2	DHS Badminton		idle	Badminton	2021-02-04 00:07:16	edit delete
3	Ball net		idle	Badminton net frame mobile portable standard multi-function air volleyball net frame 1006 multi-function air volleyball net frame	2021-02-04 00:06:57	edit delete

Figure 14. Equipment management interface.

5.9 Reservation Site Management Interface

The administrator can view, add and manage the reservation site information, and its interface is shown in Figure 15.



Booking court list

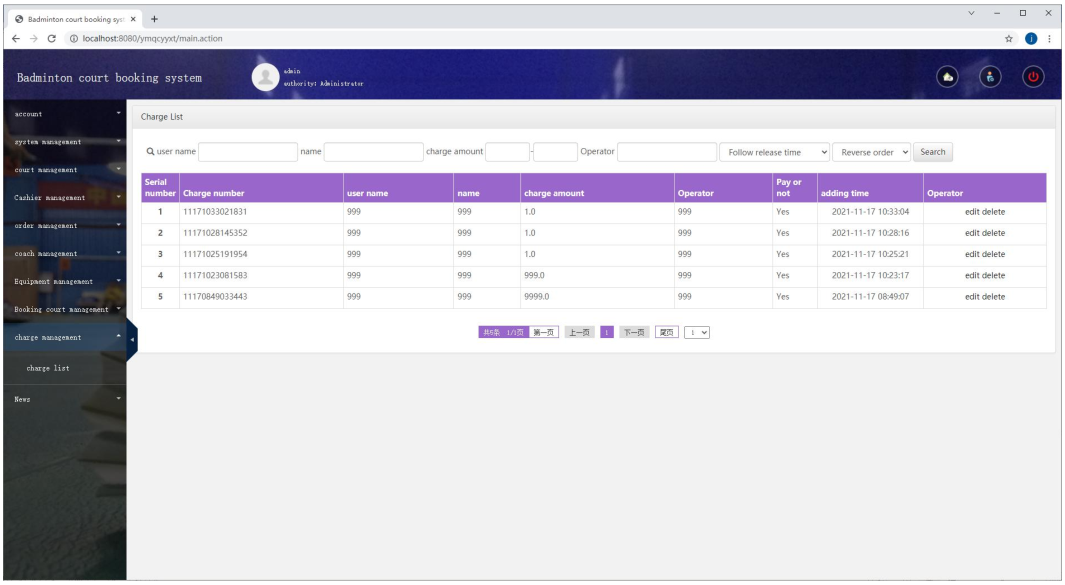
court name Booking start time - Booking leave time - Follow release time
Reversing order Seach

Serial number	Booking number	court name	price	status	Booking start time	Booking leave time	length	总额	Remark	Booking user	Pay or not	create time	Operate
1	11171106252146	Beijin badminton court	0.0	Waiting for use	2021-11-17 11:06:25	2021-11-17 12:06:25	2	0.0		999	No To pay	2021-11-17 11:06:31	search enter search leave edit delete
2	11171006137658	tokyo badminton	1000.0	Waiting for use	2021-11-17 10:06:13	2021-11-17 11:06:13	2	2000.0		999	No To pay	2021-11-17 10:22:36	search enter search leave edit delete
3	11151554511647	tokyo badminton	1000.0	Waiting for use	2021-11-15 15:54:51	2021-11-15 15:54:51	1	1000.0		999	No To pay	2021-11-15 15:54:55	search enter search leave edit delete
4	11071659551773	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:59:55	2021-11-07 16:59:55	1	1000.0		xiaochai	No To pay	2021-11-07 16:59:56	search enter search leave edit delete
5	11071659551773	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:59:55	2021-11-07 16:59:55	1	1000.0		xiaochai	No To pay	2021-11-07 16:59:56	search enter search leave edit delete
6	11071645081392	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:45:08	2021-11-07 16:45:08	1	1000.0		xiaochai	No To pay	2021-11-07 16:51:42	search enter search leave edit delete
7	11071645081392	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:45:08	2021-11-07 16:45:08	1	1000.0		xiaochai	No To pay	2021-11-07 16:48:50	search enter search leave edit delete
8	11071645081392	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:45:08	2021-11-07 16:45:08	1	1000.0		xiaochai	No To pay	2021-11-07 16:45:09	search enter search leave edit delete
9	11071642571563	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:42:57	2021-11-07 16:42:57	1	1000.0		xiaochai	No To pay	2021-11-07 16:42:58	search enter search leave edit delete
10	11071638202083	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:38:20	2021-11-07 16:38:20	1	1000.0		xiaochai	No To pay	2021-11-07 16:40:17	search enter search leave edit delete
11	11071638202083	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:38:20	2021-11-07 16:38:20	1	1000.0		xiaochai	No To pay	2021-11-07 16:38:21	search enter search leave edit delete
12	11071627112036	tokyo badminton	1000.0	Waiting for use	2021-11-07 16:27:11	2021-11-07 16:27:11	1	1000.0		xiaochai	No To pay	2021-11-07 16:27:12	search enter search leave edit delete

Figure 15 . Reservation site management interface.

5.10 Recharge Management Interface

The administrator can view, add and manage recharge information, and its interface is shown in Figure 16.

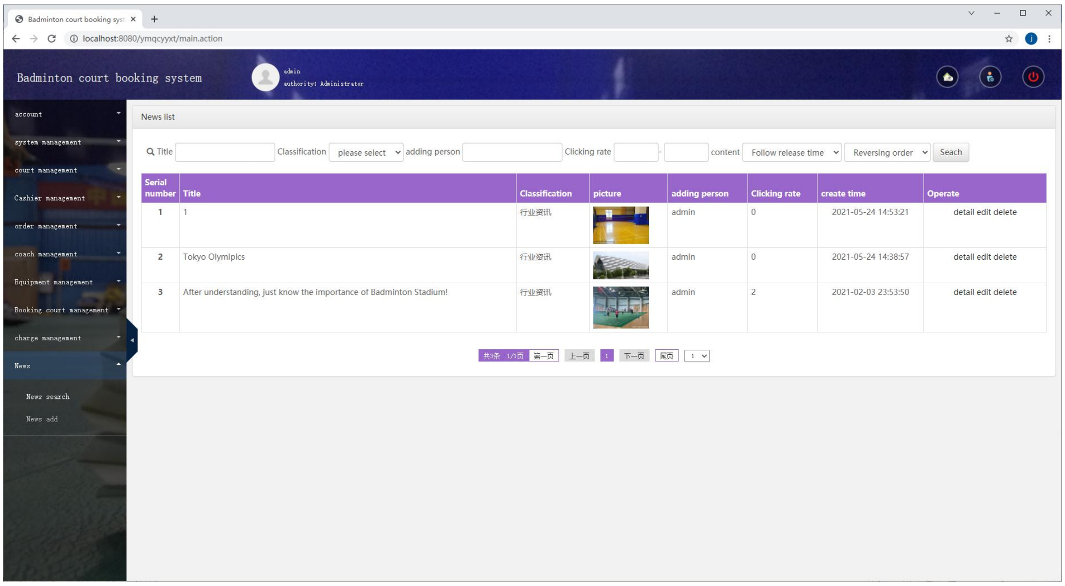


Serial number	Charge number	user name	name	charge amount	Operator	Pay or not	adding time	Operator
1	11171033021831	999	999	1.0	999	Yes	2021-11-17 10:33:04	edit delete
2	11171028145352	999	999	1.0	999	Yes	2021-11-17 10:28:16	edit delete
3	11171025191954	999	999	1.0	999	Yes	2021-11-17 10:25:21	edit delete
4	11171023081583	999	999	999.0	999	Yes	2021-11-17 10:23:17	edit delete
5	11170849033443	999	999	9999.0	999	Yes	2021-11-17 08:49:07	edit delete

Figure 16. Recharge management interface.

5.11 News Management Interface

Administrators can view, add and manage news information, and its interface is shown in Figure 17

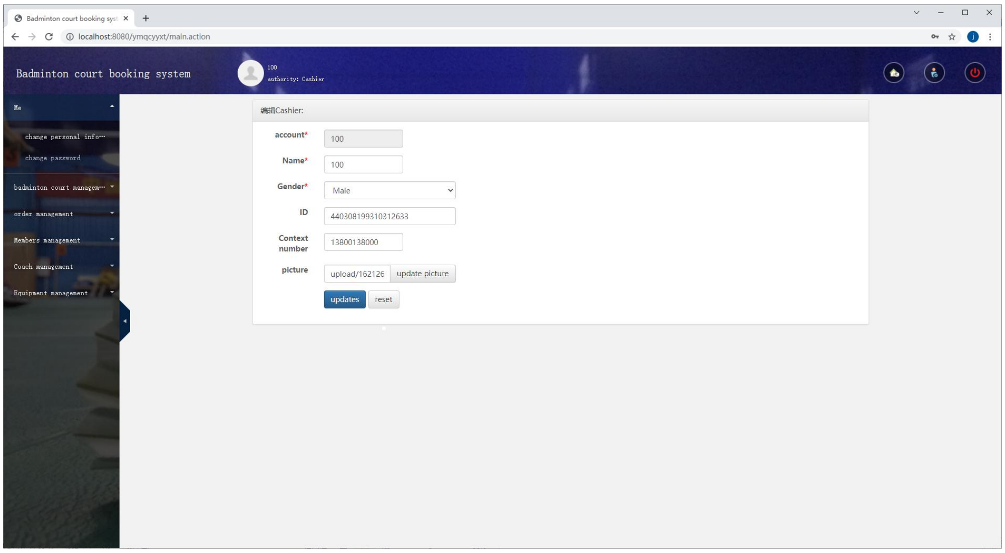


Serial number	Title	Classification	picture	adding person	Clicking rate	create time	Operate
1	1	行业资讯		admin	0	2021-05-24 14:53:21	detail edit delete
2	Tokyo Olymipics	行业资讯		admin	0	2021-05-24 14:38:57	detail edit delete
3	After understanding, just know the importance of Badminton Stadium!	行业资讯		admin	2	2021-02-03 23:53:50	detail edit delete

Figure 17. News management interface.

5.12 Cashier Interface

The cashier can change his own information in this interface, as shown in Figure 18.



编辑Cashier:

account* 100

Name* 100

Gender* Male

ID 440308199310312633

Context number 13800138000

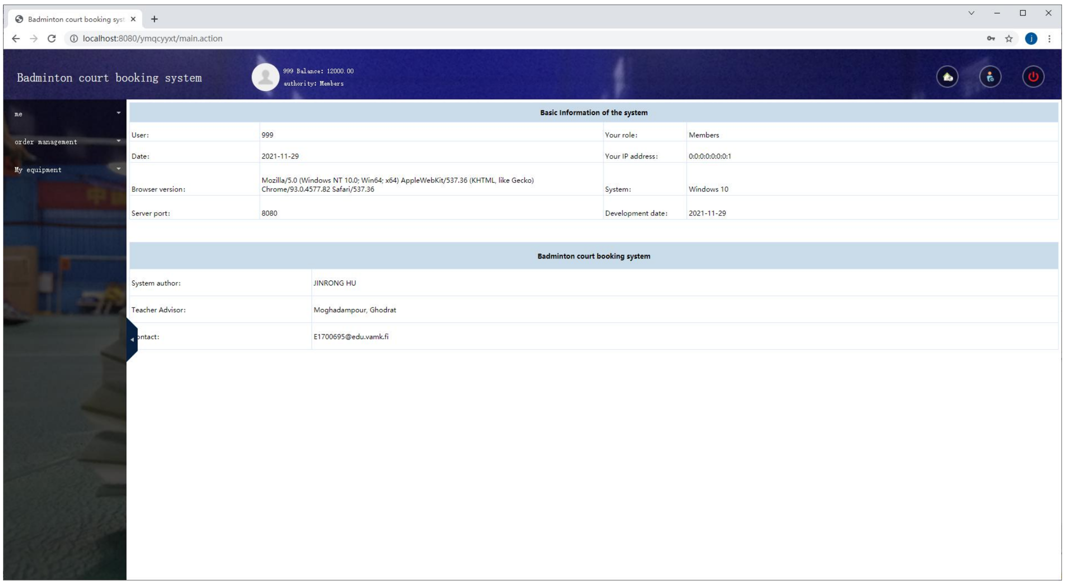
picture upload/162126 update picture

updates reset

Figure 18. Cashier interface.

5.13 User Interface

User could login this interface to manage their information and charge their account, see what equipment they have as the figure 19 show below.


Badminton court booking syst
localhost:8080/ymqcyyxt/main.action
Badminton court booking system
999 Balance: 12000.00
authority: Members
me
order management
My equipment

Basic Information of the system			
User:	999	Your role:	Members
Date:	2021-11-29	Your IP address:	0:0:0:0:0:0:0:1
Browser version:	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/93.0.4577.82 Safari/537.36	System:	Windows 10
Server port:	8080	Development date:	2021-11-29

Badminton court booking system	
System author:	JINRONG HU
Teacher Advisor:	Moghadampour, Ghodrat
ontact:	E1700695@edu.vamk.fi

Figure 19. User interface.

5.13.1 Authentication Login Code

Get the user id and password from the request and pass the data to a later method for validation.

Code snippet 2. Implementation of login function

```
public String authAdminLogin()

{

    String username = Request.get("username");

    String pwd  = Request.get("pwd");

    String cx = Request.get("cx");

    return authLoginUser(true , username,pwd,cx);

}
```

5.14 Homepage

When the user enters the system, they will first be taken to the main screen. In the main screen the user can see a row of buttons at the top of the screen. The last of these buttons is the login button for administrators and receivers. The login button for users is in the top right-hand corner of the interface. This is shown in Figure 20 below.

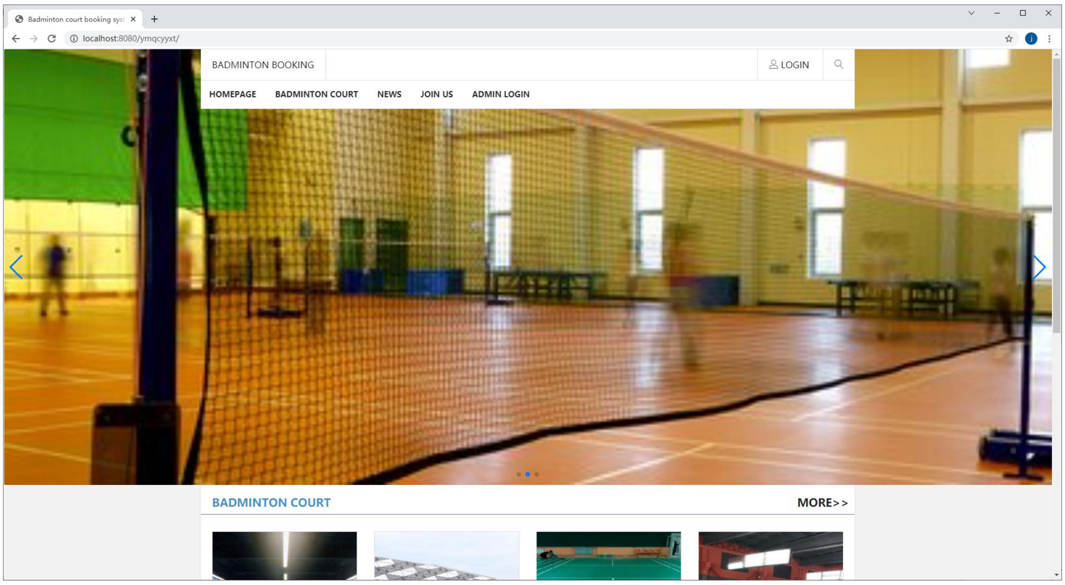


Figure 20. Homepage interface.

When the user pulls down the screen, they can see a number of tabs. This section of the page will show some of the relevant news and some of the pitches. This is shown in Figure 21 below.

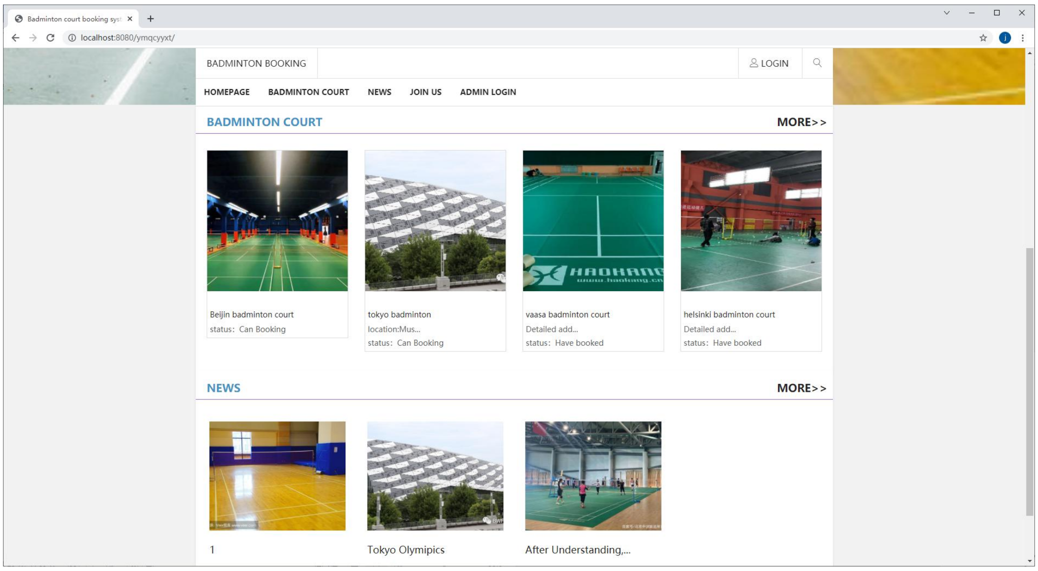


Figure 21. Homepage interface.

When the user clicks on the Badminton Hall button, they will see all the badminton halls. This is shown in Figure 22.

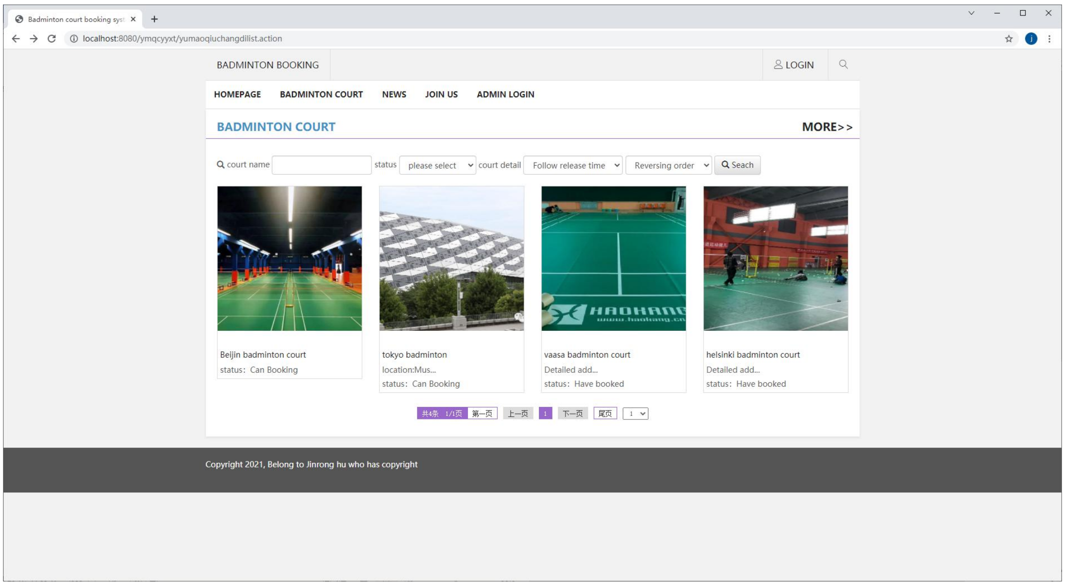
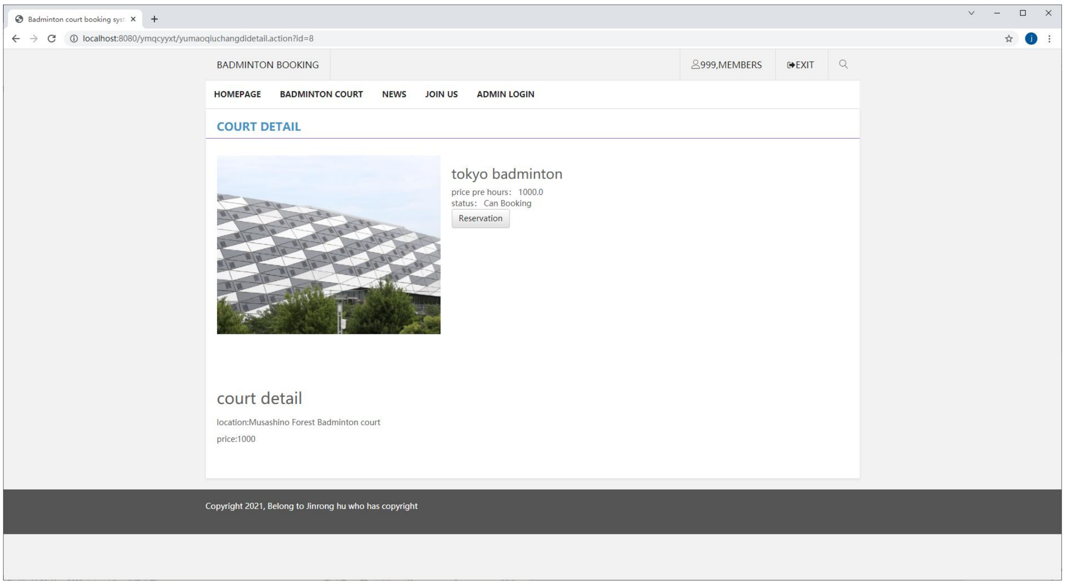


Figure 22. All badminton court interface.

When the user clicks on any badminton court, we can see its status. and the price per hour. As shown in Figure 23.



COURT DETAIL

tokyo badminton

price pre hours: 1000.0
status: Can Booking

court detail

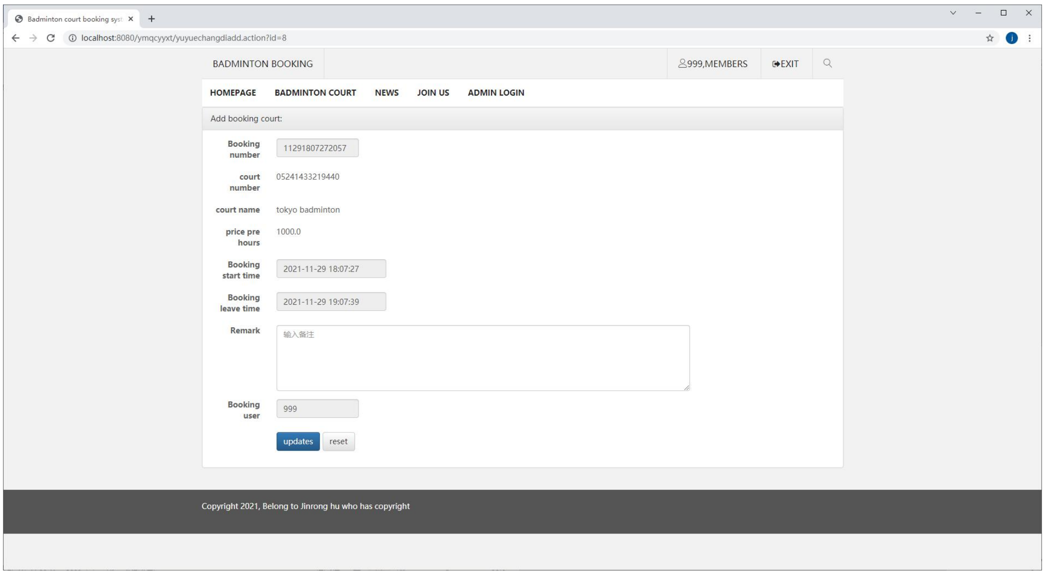
location:Musashino Forest Badminton court

price:1000

Copyright 2021, Belong to Jinrong hu who has copyright

Figure 23. badminton court interface.

When the user clicks on the booking, the system will jump to the booking screen. In this screen, the system will automatically generate a non-changeable booking number and then the client will select the entry time and departure time. The customer can add comments according to their preference and finally submit. This is shown in Figure 24.



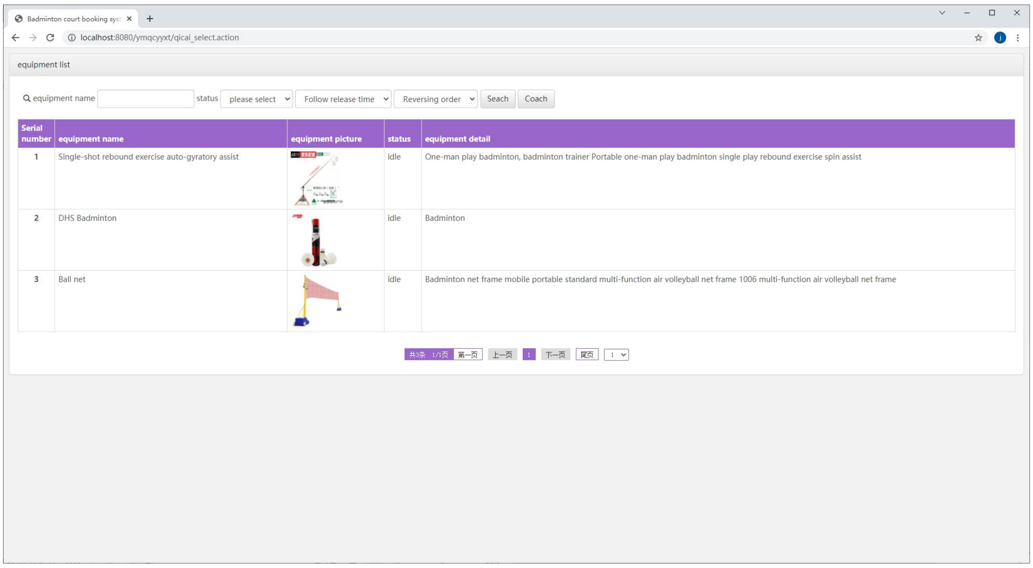
Add booking court:

Booking number	11291807272057
court number	05241433219440
court name	tokyo badminton
price pre hours	1000.0
Booking start time	2021-11-29 18:07:27
Booking leave time	2021-11-29 19:07:39
Remark	输入备注
Booking user	999

Copyright 2021, Belong to Jinrong hu who has copyright

Figure 24. Booking badminton court interface.

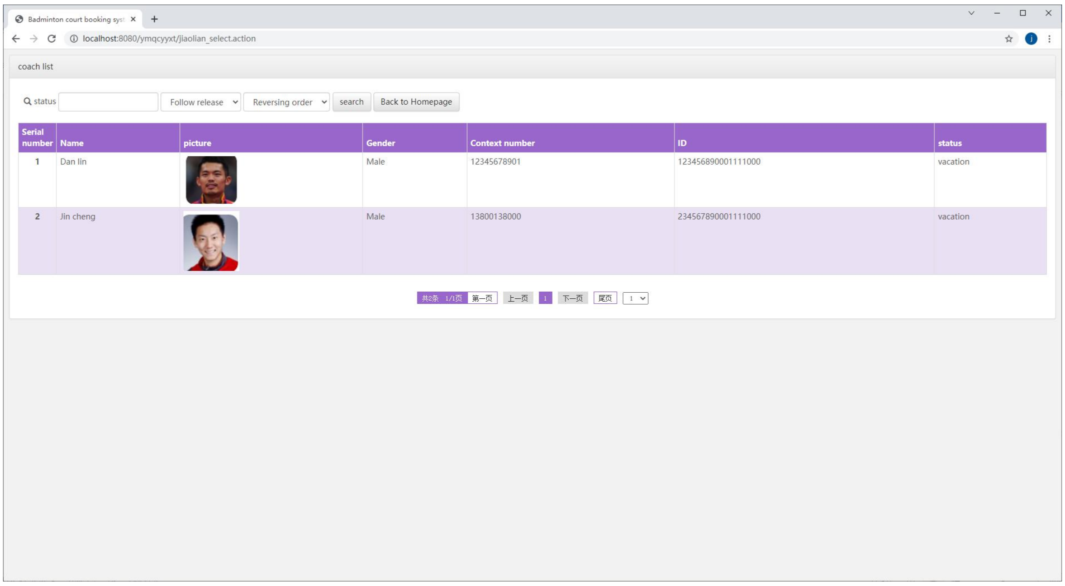
When submitted by the customer, the system will automatically jump to the equipment page, prompting the customer as to what equipment will be available at the site. This is shown in Figure 25.



Serial number	equipment name	equipment picture	status	equipment detail
1	Single-shot rebound exercise auto-gyratory assist		idle	One-man play badminton, badminton trainer Portable one-man play badminton single play rebound exercise spin assist
2	DHS Badminton		idle	Badminton
3	Ball net		idle	Badminton net frame mobile portable standard multi-function air volleyball net frame 1006 multi-function air volleyball net frame

Figure 25. Equipment interface.

At this point, the user clicks on the coach button to jump to the coach industry, and the status and introduction of the coach will be displayed on this page. This is shown in Figure 26.



Badminton court booking syst
localhost:8080/ymqcyyxt/jiaolian_select.action
coach list
status
Follow release
Reversing order
search
Back to Homepage

Serial number	Name	picture	Gender	Context number	ID	status
1	Dan lin		Male	12345678901	123456890001111000	vacation
2	Jin cheng		Male	13800138000	234567890001111000	vacation

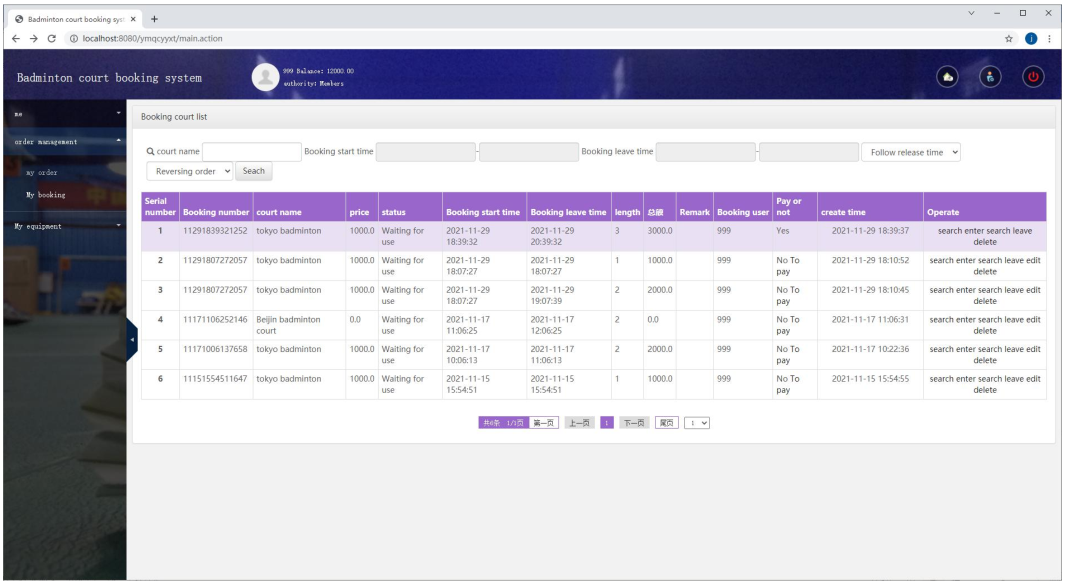
共2条 1/1页 第一页 上一页 1 下一页 尾页 1

Figure 26. Coach interface.

Finally, the user clicks on the home button to complete the booking request.

5.15 Payment Page

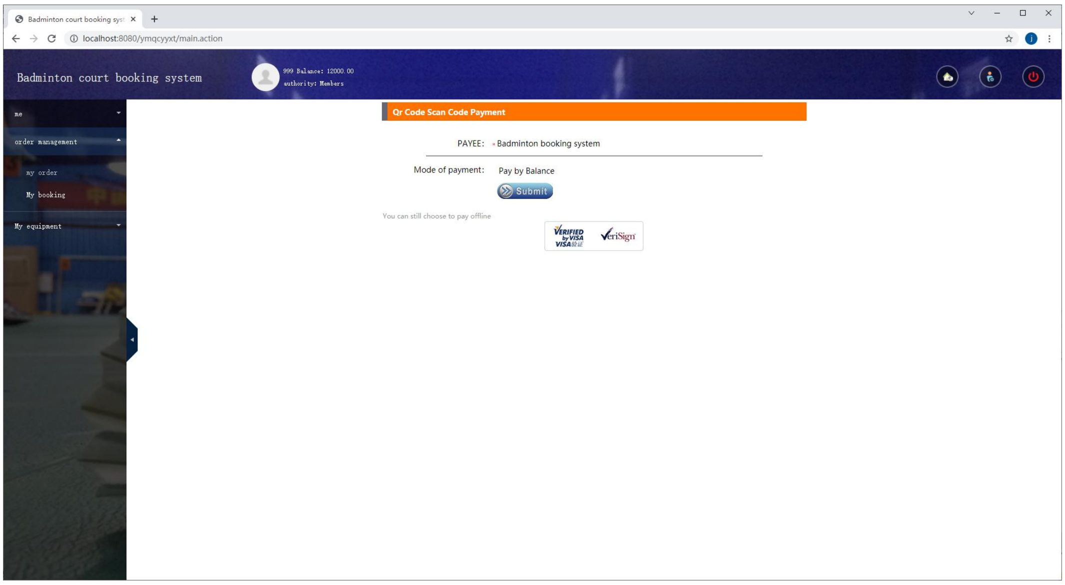
When the user enters the home page backstage, click on My Orders, you can see that the payment status is None and the user can choose to pay. As shown in Figure 27.



Serial number	Booking number	court name	price	status	Booking start time	Booking leave time	length	总额	Remark	Booking user	Pay or not	create time	Operate
1	11291839321252	tokyo badminton	1000.0	Waiting for use	2021-11-29 18:39:32	2021-11-29 20:39:32	3	3000.0		999	Yes	2021-11-29 18:39:37	search enter search leave delete
2	11291807272057	tokyo badminton	1000.0	Waiting for use	2021-11-29 18:07:27	2021-11-29 18:07:27	1	1000.0		999	No To pay	2021-11-29 18:10:52	search enter search leave edit delete
3	11291807272057	tokyo badminton	1000.0	Waiting for use	2021-11-29 18:07:27	2021-11-29 19:07:39	2	2000.0		999	No To pay	2021-11-29 18:10:45	search enter search leave edit delete
4	11171106252146	Beijin badminton court	0.0	Waiting for use	2021-11-17 11:06:25	2021-11-17 12:06:25	2	0.0		999	No To pay	2021-11-17 11:06:31	search enter search leave edit delete
5	11171006137658	tokyo badminton	1000.0	Waiting for use	2021-11-17 10:06:13	2021-11-17 11:06:13	2	2000.0		999	No To pay	2021-11-17 10:22:36	search enter search leave edit delete
6	11151554511647	tokyo badminton	1000.0	Waiting for use	2021-11-15 15:54:51	2021-11-15 15:54:51	1	1000.0		999	No To pay	2021-11-15 15:54:55	search enter search leave edit delete

Figure 27. Order state interface.

The user then clicks on Pay and jumps to the payment page. It will prompt us to pay with our balance. This is shown in Figure 28.


Badminton court booking syst
localhost:8080/ymqcyyxt/main.action
Badminton court booking system
me
order management
my order
My booking
My equipment
Submit
VERIFIED by VISA VISA验证
VeriSign

999 Balance: 12000.00
authority: Members

Qr Code Scan Code Payment

PAYEE: * Badminton booking system

Mode of payment: Pay by Balance

You can still choose to pay offline

Figure 28. Payment interface.

The balance will need to be recharged offline.

6 TESTING

The final stage in the development of the system is system testing. This is crucial to the overall system because the quality of testing is closely linked to the development of the product. Also, the quality, performance and reliability of the software required by the customer are achieved through testing. The entire testing process must follow the principles of rigor, perfection and standardization. The main purpose of testing is to detect any errors in the operation of the system and then to debug the errors until the program works perfectly. In practice, software testing can only minimize errors but not eliminate them. However, it is true that the fewer errors there are, the lower the probability that the system will go wrong and the easier and safer it will be for users to use. There are many types of system testing, such as functional testing, security testing, usability testing, performance testing and compatibility testing.

6.1 Test environment and Conditions

Processor: PC terminal: Intel Core Processor I3

Memory: PC terminal: 4GB

Hard disk: PC terminal: above 80G

Operating system: PC terminal: win7-10 or Windows XP

Database: MySQL

6.2 Function Testing

There are various types of functional tests, which usually include: security testing, compliance testing, suitability testing, accuracy testing and operability testing. Functional testing of the system is illustrated in Table 3.

Table 3. Website function test.

Test content	operability	accuracy	applicability	compliance	security
Test result	good	good	good	good	good

6.3 Security Test

Security testing is the most important part of the entire testing process. The quality of security is directly related to the security of user data and product information. It examines whether external intruders can gain access to the system by various means and steal or corrupt the system data. (Hsiao, Brusilovsky.2017) Therefore, this system uses ARP attacks, intercepting packets and other methods. The security testing of this system was completed based on the following steps.

1）When checking the results, the system must enter through the corresponding account and password.

2）This website has encrypted the information of all users and administrators as well as the database, which can be backed up, with high security.

6.4 Usability Test

Usability testing is employed to test the comprehensibility, operability and learnability of websites. The specific test aspects are shown in table 4.

Table 4. Website usability test.

Test items	Tester evaluation
If the operation of window moving, changing size and closing are normal?	good
If the operation module is friendly?	good
If the text description of module and prompt content are correct?	good
If the module layout is reasonable and coordinated?	good
If the status of the module is correct?	good
If keyboard and mouse operation are supported?	good
If the required data items are displayed correctly?	good
If the operation process is reasonable?	good

6.5 Performance Testing

Performance testing is the simulation of a website's operating environment to test whether the site's performance meets the client's requirements. Important technical indicators for performance testing include the speed of the website, the response time of the network and the number of connected nodes.

1） System running speed. This refers to the speed of running the system on different configurations of computers and different systems without any stutters or delays.

2） System response time. The system response time includes three main aspects: the minimum response time of the system; the average response time of the system, and the maximum response time of the system. After testing, under normal circumstances, the three values of LAN response time were. 2/3/5S, which was very fast and satisfied the users.

3）The number of concurrent nodes supported by the system, i.e. whether there is a delay in the system when the number of accesses increases. After testing, there was a small delay in access when the volume of data exceeded 5000.

6.6 Case Testing

Case testing includes many aspects. This system tests user login cases which were tested in three groups: entering the correct account number and the wrong password, entering the wrong account number and password, and entering an empty message to see if the system could be accessed. According to the test results, the system failed to log in in all three cases, so the test was successful.

6.7 Analysis of Test Results

In terms of the design, this badminton court booking system draws on the best systems at home and abroad, from the interface to the system design, to ensure that administrators and users can easily operate it. Its main features and advantages are summarized below:

（1）The system is highly portable and targeted. The high degree of relevance allows a better service, and the portability facilitates the operation on multiple systems, which brings great convenience to the customer.

（2） The badminton court booking system is comprehensive and easy to manage. It deals with all kinds of errors and exceptions in a timely and comprehensive manner, avoiding many errors caused by inadvertent customer operations. Because this system is easy to operate and has a user-friendly interface, it can be operated well by anyone with internet access.

After an overall testing and analysis, the design and implementation of the badminton court booking system proved to be able to meet the requirements and needs of the client. The product is fully functional, simple to operate, and demonstrates good operational performance. Therefore, it is capable of promoting the information management of the badminton court reservation system and has huge potential to be developed in the future.

7 CONCLUSION

This system illustrates the feasibility of the badminton court booking system from both hardware and software counterparts through an introduction to the Java language and MySQL database. The conclusions and findings of this paper are that a badminton court booking system built with the Java language and MySQL can be implemented and the website can be displayed responsively.

The system basically fulfill the objectives envisaged for each section. The administrator can manage essentially all functions. This includes equipment functions, venue functions, coaching functions, and information management for all members. One of the most challenging features is the banner function. The cashier has management rights for a portion of the functions. The most important thing for the cashier is that he/she can recharge the customers as well as manage the orders of the users. For customers, the most important functions of this system are online booking of badminton courts and news viewing.

7.1 Future Work

It is a great pity that the system was designed to be in Chinese in the first place. This is due to the fact that many plugins cannot be found and used in the system due to the fact that China has blocked most websites from other countries. For example, the timetable plug-in remains in the system as a Chinese plug-in.

The escort function has been taken into account in the design of the system. The role of a chaperone is different from that of a coach. The role is designed for those who are alone but want to play badminton. This is because badminton must be played by two people. If a court offers this service to unaccompanied customers, it is possible to charge an additional fee. However, this feature is not available on all courts and has not been included in the system.

This system can be revised and improved in the future. This system could be used in a tabletop card game shop run by the author. Customers will still be able to book rooms and the card games they want to play by making reservations online. Customers will book online and pay offline. A news section can be used to advertise new themed rooms as well as themed games. Tabletop card games are a very popular mode of socializing at the moment. Young people are keen on this type of game which is both competitive and social.

8 REFERENCE

baidu.com .2021. JDBC 5thDecember. https://baike.baidu.com/item/Java 数据库连接?fromtitle=jdbc&fromid=485214

baidu.com.2021,tomcat,5thDecember.https://baike.baidu.com/item/tomcat/255751?fr=aladdin.

computertechreviews.com. Advantage of web application. Access 23 December 2021. What is Web Application? - Definition, Advantages, Examples, And More (computertechreviews.com)

He Dinghua. Java programming case tutorial [M]. Beijing: Tsinghua University Press, 2019.

Liu, Chunmao. Java MySQL dynamic web development cases classroom [M]. Beijing: Tsinghua University Press, 2018.

Lie Xu song, Chen W. Core technologies and best practices of Java (2nd ed.) [M]. Beijing: Mechanical Industry Press, 2018.

Lin Shixin. Java programming fundamentals tutorial [M]. Beijing: Electronic Industry Press, 2018.

segmentfault.com. 2021. Introduction of LayUI 5December.

Tomorrow's Technology. Java project development throughout the whole process [M]. Beijing: Tsinghua University Press, 2018.

Java Dao model. 5December. 2021. www.runoob.com.

Xiao Ying. Solving the Chinese garbled code problem in Java/servlet development[J]. Science and technology communication, 2017, (1)11-25.

9 APPENDICES

9.1 Appendices 1 Admin Core Code

```
    public String getWhere()
    {
        String where = " ";


                        if(!Request.get("username").equals("")) {
            where += " AND username LIKE '%"+Request.get("username")+"%' 
";
            }
                return where;
    }

        /**
     * add
     * @return
     */
    public String AdminAdd()
    {
        if(!checkLogin()){
            return showError("please login" , "login.action");
        }
                return success;
    }

    /**
     *  modified
     */
    public String updt()
    {
        int id = Request.getInt("id");
        Admins mmm = dao.find(id);
        request.setAttribute("mmm" , mmm);
        request.setAttribute("updtself" , 0);
        return success;
    }

    public String updtself()
    {
        int id = (int)request.getSession().getAttribute("id");
        Admins mmm = dao.find(id);
        request.setAttribute("mmm" , mmm);
        request.setAttribute("updtself" , 1);
        return success;
    }
    /**
     * add data
     * @return
     */
    public String insert()
    {

        if(request.getParameter("login")!=null && !checkLogin()){
            return showError("You have not logged in please log in after
the operation");
        }
        String tmp="";
        Admins post = new Admins();
                            post.setUsername(Request.get("username"));
                                    post.setPwd(Request.get("pwd"));

post.setAddtime(Info.getDateStr());
                dao.insert(post);
        int charuid = post.getId().intValue();
                return showSuccess("add success" ,
Request.get("referer").equals("")?request.getHeader("referer"):Request.ge
t("referer"));
    }

    /**
     * update data
     * @return
     */
    public String update()
    {
        if(request.getParameter("login")!=null && !checkLogin()){
            return showError("You have not logged in please log in after
the operation");
        }

        Admins post = dao.find(Request.getInt("id"));

if(!Request.get("username").equals(""))
                post.setUsername(Request.get("username"));

if(!Request.get("pwd").equals(""))
                post.setPwd(Request.get("pwd"));

        post.setId(Request.getInt("id"));
```

```
                dao.update(post,Request.getInt("id"));
        int charuid = post.getId().intValue();

        if(Request.getInt("updtself") == 1){
            return showSuccess("Preservation successful" ,
"admins_updtself.action");
        }
        return showSuccess("Update successful" ,
Request.get("referer").equals("")?request.getHeader("referer"):Request.ge
t("referer"));
    }

    public String delete()
    {
        if(!checkLogin())
        {
            return showError("You have not logged in please log in after
the operation");
        }
        int id = Request.getInt("id");
        dao.delete(id);
        return showSuccess("Deletion successful" ,
request.getHeader("referer"));
    }


    public AdminsDao getDao() {
        return dao;
    }

    public void setDao(AdminsDao dao) {
        this.dao = dao;
    }


}
```

9.2 Appendices 2 Upload Core Code

```
package com.action;


import com.common.BaseAction;

import org.apache.struts2.ServletActionContext;

import util.Request;


import Java.io.*;

import Java.util.Date;

import Java.util.Map;

import Java.util.Random;


/**

 * upload files

 */

public class Upload extends BaseAction {

    private static final int BUFFER_SIZE = 16 * 1024;


    private File fujian;

    private String fujianFileName;
```

```
  private String fujianContentType;


  /**
   * upload files work way
   * @return
   */
  public String upload()
  {
     Random random = new Random();


     String newFujianName=new Date().getTime()+String.valueOf(Math.abs(random.nextInt())).substring(0,5)+fujianFileName.substring(fujianFileName.indexOf("."));
     String dstPath = ServletActionContext.getServletContext().getRealPath("upload")+ "\\" + newFujianName;
     File dstFile = new File(dstPath);
     File dir = new File(dstFile.getParent());


     if(!dir.exists())
     {
        dir.mkdirs();
        }
        copy(this.getFujian(),dstFile);
        request.setAttribute("url", "upload"+ "/" + newFujianName);
        return success;
    }


    /**
     * umeditor  deal with the word on the editor
     * @return
     */
    public String umeditor()
    {
        upload();
        String url = (String) request.getAttribute("url");
        String result = "{\"name\":\"" + fujian.getName() + "\", \"originalName\": \"" + fujian.getName() + "\", \"size\": " + fujian.length() + ", \"state\": \"success\", \"type\": \"" + fujianContentType + "\", \"url\": \"" + url + "\"}";


        out(result);
        return null;
```

```
}

/**
 * copy files to the right place
 * @param src
 * @param dst
 */
private static void copy(File src, File dst)
{
   InputStream in = null;
   OutputStream out = null;
   try
   {
      in = new BufferedInputStream(new FileInputStream(src), BUFFER_SIZE);
      out = new BufferedOutputStream(new FileOutputStream(dst),BUFFER_SIZE);
      byte[] buffer = new byte[BUFFER_SIZE];
      int len = 0;
      while ((len = in.read(buffer)) > 0)
      {
         out.write(buffer, 0, len);
   }
}
catch (Exception e)
{
   e.printStackTrace();
}
finally
{
   if (null != in)
   {
      try
      {
         in.close();
      }
      catch (IOException e)
      {
         e.printStackTrace();
      }
   }
   if (null != out)
```

```
        {
            try
            {
                out.close();
            }
            catch (IOException e)
            {
                e.printStackTrace();
            }
        }
    }
}

public File getFujian() {
    return fujian;
}

public void setFujian(File fujian) {
    this.fujian = fujian;
}
```

```
    public String getFujianFileName() {

        return fujianFileName;

    }



    public void setFujianFileName(String fujianFileName) {

        this.fujianFileName = fujianFileName;

    }



    public String getFujianContentType() {

        return fujianContentType;

    }



    public void setFujianContentType(String fujianContentType) {

        this.fujianContentType = fujianContentType;

    }

}
```